

WHAT IS CLAIMED IS:

1 1. A method for loading software on a plurality of
2 processors in a heterogeneous processor environment,
3 said method comprising:
4 retrieving a file using a first processor;
5 detecting a processor identifier that corresponds to
6 the file;
7 determining whether to load the file on a second
8 processor based upon the processor identifier; and
9 loading the file onto the second processor in response
10 to the determination.

1 2. The method as described in claim 1 further comprising:
2 executing a program on the first processor;
3 loading a runtime loader onto the first processor in
4 response the execution; and
5 performing the retrieving, detecting, and the
6 determining using the runtime loader.

1 3. The method as described in claim 1 wherein the file is
2 an executable file.

3 4. The method as described in claim 3 further comprising:
4 sending a plug-in to the second processor using the
5 first processor, the plug-in corresponding to the
6 file;
7 sending data to the second processor using the first
8 processor, the data corresponding to the plug-in; and

9 processing the data with the plug-in using the second
10 processor.

1 5. The method as described in claim 3 further comprising:
2 retrieving a plug-in using the second processor, the
3 plug-in corresponding to the file;

4 retrieving data using the second processor, the data
5 corresponding to the plug-in; and

6 processing the data with the plug-in using the second
7 processor.

1 6. The method as described in claim 3 wherein the
2 executable file is in a file format, and wherein the
3 file format is selected from the group consisting of
4 an ELF format, an XCOFF format, and a PE/COFF format.

1 7. The method as described in claim 1 wherein the
2 processor identifier is a machine type, the
3 determining further comprising:
4 extracting the machine type from the file; and
5 comparing the machine type to a plurality of machine
6 types.

1 8. The method as described in claim 1 wherein the file is
2 part of a combined file, and wherein the processor
3 type corresponds to one or more section headers from a
4 plurality of section headers.

1 9. The method as described in claim 1 wherein the file is
2 part of a combined file, and wherein the combined file

3 includes one or more processor identifiers that
4 correspond to the first processor.

1 10. The method as described in claim 1 wherein the first
2 processor is a processing unit and wherein the second
3 processor is a synergistic processing unit.

1 11. An information handling system comprising:
2 a plurality of processors in a heterogeneous processor
3 environment;

4 a memory accessible by the plurality of processors;

5 one or more nonvolatile storage devices accessible by
6 the plurality of processors; and

7 a software loading tool for loading software on a
8 plurality of processors, the software loading tool
9 comprising software code effective to:

10 retrieve a file using a first processor from
11 one of the nonvolatile storage devices;

12 detect a processor identifier using the
13 first processor that corresponds to the
14 file;

15 determine whether to load the file on a
16 second processor based upon the processor
17 identifier; and

18 load the file onto the second processor in
19 response to the determination.

1 12. The information handling system as described in claim
2 11 wherein the software code is further effective to:
3 execute a program on the first processor;

4 load a runtime loader onto the first processor in
5 response the execution; and
6 perform the retrieving, detecting, and the determining
7 using the runtime loader located on the first
8 processor.

1 13. The information handling system as described in claim
2 11 wherein the file is an executable file.

3 14. The information handling system as described in claim
4 13 wherein the software code is further effective to:
5 send a plug-in to the second processor using the first
6 processor, the plug-in corresponding to the file;
7 send data to the second processor using the first
8 processor, the data corresponding to the plug-in; and
9 process the data with the plug-in using the second
10 processor.

1 15. The information handling system as described in claim
2 13 wherein the software code is further effective to:
3 retrieve a plug-in using the second processor from one
4 of the nonvolatile storage devices, the plug-in
5 corresponding to the file;
6 retrieve data using the second processor from one of
7 the nonvolatile storage devices, the data
8 corresponding to the plug-in; and
9 process the data with the plug-in using the second
10 processor.

1 16. The information handling system as described in claim
2 13 wherein the executable file is in a file format,
3 and wherein the file format is selected from the group
4 consisting of an ELF format, an XCOFF format, and a
5 PE^{COFF} format.

1 17. The information handling system as described in claim
2 11 wherein the processor identifier is a machine type,
3 and wherein the software code is further effective to:
4 extract the machine type from the file; and
5 compare the machine type to a plurality of machine
6 types.

1 18. The information handling system as described in claim
2 11 wherein the file is part of a combined file, and
3 wherein the processor type corresponds to one or more
4 section headers from a plurality of section headers.

1 19. The information handling system as described in claim
2 11 wherein the file is part of a combined file, and
3 wherein the combined file includes one or more
4 processor identifiers that correspond to the first
5 processor.

1 20. The information handling system as described in claim
2 11 wherein the first processor is a processing unit
3 and wherein the second processor is a synergistic
4 processing unit.

1 21. A computer program product stored on a computer
2 operable media for loading software on a plurality of

3 processors in a heterogeneous processor environment,
4 said computer program product comprising:
5 means for retrieving a file using a first processor;
6 means for detecting a processor identifier that
7 corresponds to the file;
8 means for determining whether to load the file on a
9 second processor based upon the processor identifier;
10 and
11 means for loading the file onto the second processor
12 in response to the determination.

1 22. The computer program product as described in claim 21
2 further comprising:
3 means for executing a program on the first processor;
4 means for loading a runtime loader onto the first
5 processor in response the execution; and
6 means for performing the retrieving, detecting, and
7 the determining using the runtime loader.

1 23. The computer program product as described in claim 21
2 wherein the file is an executable file.

3 24. The computer program product as described in claim 23
4 further comprising:
5 means for sending a plug-in to the second processor
6 using the first processor, the plug-in corresponding
7 to the file;
8 means for sending data to the second processor using
9 the first processor, the data corresponding to the
10 plug-in; and

11 means for processing the data with the plug-in using
12 the second processor.

1 25. The computer program product as described in claim 23
2 further comprising:
3 means for retrieving a plug-in using the second
4 processor, the plug-in corresponding to the file;
5 means for retrieving data using the second processor,
6 the data corresponding to the plug-in; and
7 means for processing the data with the plug-in using
8 the second processor.

1 26. The computer program product as described in claim 23
2 wherein the executable file is in a file format, and
3 wherein the file format is selected from the group
4 consisting of an ELF format, an XCOFF format, and a
5 PECOFF format.

1 27. The computer program product as described in claim 21
2 wherein the processor identifier is a machine type,
3 the means for determining further comprising:
4 means for extracting the machine type from the file;
5 and
6 means for comparing the machine type to a plurality of
7 machine types.

1 28. The computer program product as described in claim 21
2 wherein the file is part of a combined file, and
3 wherein the processor type corresponds to one or more
4 section headers from a plurality of section headers.

1 29. The computer program product as described in claim 21
2 wherein the file is part of a combined file, and
3 wherein the combined file includes one or more
4 processor identifiers that correspond to the first
5 processor.

1 30. The computer program product as described in claim 21
2 wherein the first processor is a processing unit and
3 wherein the second processor is a synergistic
4 processing unit.